

droplet

February 13, 2023

1 Droplets

1.1 Different morphologies of the droplet state

Let us consider the equilibrium droplet state, *i.e.*, $\alpha < 0$ and $-\sqrt{\frac{-\alpha}{\beta}} < \phi_0 < \sqrt{\frac{-\alpha}{\beta}}$. For a fixed value of α , depending on the global average density ϕ_0 , we can have 3 different types of droplet state. For $\phi_0 < 0$, we have a liquid droplet surrounded by vapour phase. For $\phi_0 > 0$, we have a vapour bubble surrounded by liquid phase. Finally for $\phi_0 = 0$, the liquid droplet forms a strip, as we can see from the figure below.

```
[ ]: import numpy as np
import matplotlib.pyplot as plt
import matplotlib.animation as animation

# parameters
dx = 1.0 # normally dx = dy
dt = 0.01

Nx = 32 # normally the total number of lattice = 2^(some integer)
Ny = 32 # this is required for Fourier transform later
Nt = 1000000

alpha, beta, kappa = -1.0, 1.0, 1.0
phi0 = 0.0

# array of cartesian coordinates (needed for plotting)
x = np.arange(0, Nx)*dx
y = np.arange(0, Ny)*dx
y, x = np.meshgrid(y, x)

# create a phi-matrix of size Nx by Ny (similarly for mu-matrix)
phi = np.zeros((Nx, Ny)) # phi at current timestep
mu = np.zeros((Nx, Ny))

# method to calculate the laplacian
def laplacian(phi):
    # axis=0 --> roll along x-direction
    # axis=1 --> roll along y-direction
```

```

laplacianphi = \
    (np.roll(phi,+1,axis=0) - 2.0*phi + np.roll(phi,-1,axis=0))/(dx*dx) \
    + (np.roll(phi,+1,axis=1) - 2.0*phi + np.roll(phi,-1,axis=1))/(dx*dx)

    return laplacianphi

# takes in phi at time n and outputs phi at time n+1
def update(phi):
    mu = alpha*phi + beta*phi*phi*phi - kappa*laplacian(phi)
    phi = phi + dt*laplacian(mu)

    return phi

# plot configurations of phi0, phi1, phi2
def plot(phi0, phi1, phi2, label0, label1, label2):
    # initialize figure and movie objects
    fig, (ax1, ax2, ax3) = plt.subplots(1,3,figsize=(6,2))

    ax1.set_title(label0)
    ax1.set_aspect('equal')
    colormap = ax1.pcolormesh(x, y, phi0, shading='auto', vmin=-1.2, vmax=1.2)

    ax2.set_title(label1)
    ax2.set_aspect('equal')
    colormap = ax2.pcolormesh(x, y, phi1, shading='auto', vmin=-1.2, vmax=1.2)

    ax3.set_title(label2)
    ax3.set_aspect('equal')
    colormap = ax3.pcolormesh(x, y, phi2, shading='auto', vmin=-1.2, vmax=1.2)

    plt.show()

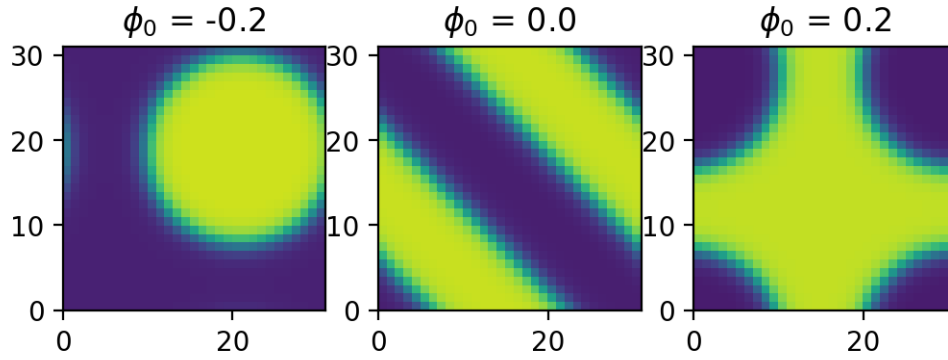
# run simulation for Nt timesteps for given phi0
def run(phi0):
    # initialization
    phi = phi0*np.ones((Nx, Ny)) + np.random.normal(0.0, 0.001, (Nx, Ny))

    for n in range(0, Nt, 1):
        phi = update(phi)

    return phi

plot(run(-0.2), run(0.0), run(0.2), "$\phi_0$ = -0.2", "$\phi_0$ = 0.0",
    ↪ "$\phi_0$ = 0.2")

```



1.2 Flat interface ($d = 1$ dimension)

As we can see from above, for $\phi_0 = 0$, the droplet is a strip, and the interface between the liquid and the gas phase is completely flat, reducing the problem to a one-dimensional problem $\phi(x, t)$. In the steady (or equilibrium) state $t \rightarrow \infty$, the density profile might look something like the first plot in the figure below:

```
[ ]: import numpy as np
import matplotlib.pyplot as plt

fig, ax = plt.subplots(figsize=(6,3))

alpha, beta, kappa = -1.0, 1.0, 1.0
xi = np.sqrt(2*kappa/-alpha)
x = np.arange(-20, 20, 0.001)
phi = np.tanh(x/xi)
dphidx = 1/(xi*np.cosh(x/xi)**2)
f_density = 0.5*alpha*phi**2 + 0.25*phi**4 + 0.25*kappa*dphidx**2

ax.set_xlabel('$x$')
ax.set_ylabel('$\phi(x)$')
ax.set_xlim(-20,20)

ax.plot(x, phi)

ax.annotate('', c='black',
            xy=(-2.0,0.0), # location of the end point
            xytext=(2.0,0.0), # location of the start point
            arrowprops=dict(edgecolor='black', facecolor='red',
            ↪arrowstyle='<->'))

ax.annotate('$\xi$', c='black', fontsize=12, xy=(-3,0.0))
```

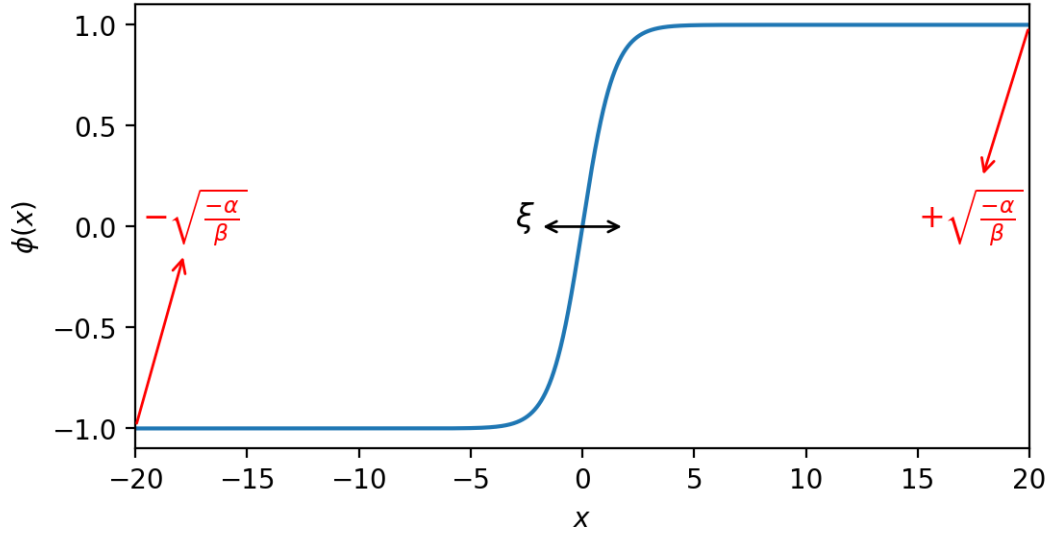
```

ax.annotate('$-\sqrt{\frac{-\alpha}{\beta}}$', c='red', fontsize=12,
            xy=(-20,-1.0), xytext=(-19.7,0.0),
            arrowprops=dict(edgecolor='red', facecolor='red', arrowstyle='<-'))

ax.annotate('$+\sqrt{\frac{-\alpha}{\beta}}$', c='red', fontsize=12,
            xy=(20,1.0), xytext=(15,0.0),
            arrowprops=dict(edgecolor='red', facecolor='red', arrowstyle='<-'))

plt.show()

```



The plot above shows the typical density field $\phi(x)$. At $x = -\infty$, we have the gas phase, where $\phi = -\sqrt{\frac{-\alpha}{\beta}}$ and at $x = +\infty$, we have the liquid phase, where $\phi = \sqrt{\frac{-\alpha}{\beta}}$. We then have an interface, assumed to be located at the origin $x = 0$, which separates the liquid from the gas phase. As we can see, the interface is not sharp, but rather is spread across some interfacial width ξ . In real physical situations, the interfacial width is typically a few molecular lengths, however in simulations, we usually use a much wider interfacial width for numerical stability.

1.2.1 Interfacial profile $\phi(x)$ and interfacial width ξ

In this subsection, we will derive the interfacial profile $\phi(x)$ and the interfacial width ξ for a flat interface, which is effectively a one-dimensional (1d) system.

In 1d, the Hamiltonian (energy) can be written as:

$$H[\phi] = A \int_{-\infty}^{\infty} dx \underbrace{\left\{ \frac{\alpha}{2} \phi^2 + \frac{\beta}{4} \phi^4 + \frac{\kappa}{2} |\nabla \phi|^2 \right\}}_{\text{1d energy density}}, \quad (1)$$

where A is the area of the system along y and z direction. (In 1d, the system is translationally invariant along y and z .) The 1d energy density consists of the local term: $\frac{\alpha}{2} \phi^2 + \frac{\beta}{4} \phi^4 = f(\phi)$,

which we will call $f(\phi)$, and the semi-local term $\frac{\kappa}{2}|\nabla\phi|^2$.

The steady state density, is given by the solution to the equation:

$$\mu(x) = \frac{\delta H}{\delta \phi} = 0 \quad \Rightarrow \quad \frac{df}{d\phi} - \kappa \frac{d^2\phi}{dx^2} = 0, \quad (2)$$

where $f(\phi) = \frac{\alpha}{2}\phi^2 + \frac{\beta}{4}\phi^4$ is the local energy density. To solve the above equation, we multiply both sides of the equation by $d\phi/dx$ to get:

$$\frac{df}{dx} - \kappa \frac{d^2\phi}{dx^2} \frac{d\phi}{dx} = 0 \quad \Rightarrow \quad \frac{df}{dx} - \frac{\kappa}{2} \frac{d}{dx} \left(\frac{d\phi}{dx} \right)^2 = 0 \quad (3)$$

Now we can integrate with respect to x to get the **magic equation**:

$$f(\phi) - \frac{\kappa}{2} \left(\frac{d\phi}{dx} \right)^2 = C, \quad (4)$$

where C is the constant of integration. The constant of integration can be found by substituting $x = \infty$ to get $C = f\left(\sqrt{\frac{\alpha}{\beta}}\right) = -\frac{\alpha^2}{4\beta}$. After rearranging, we get:

$$\frac{d\phi}{\sqrt{f(\phi) + \frac{\alpha^2}{4\beta}}} = \sqrt{\frac{2}{\kappa}} dx, \quad (5)$$

which we can integrate from $x = 0$ to x .

$$\int_0^{\phi(x)} \frac{d\phi}{\sqrt{f(\phi) + \frac{\alpha^2}{4\beta}}} = \sqrt{\frac{2}{\kappa}} x \quad (6)$$

Note that we can factorize:

$$\underbrace{\frac{\alpha}{2}\phi^2 + \frac{\beta}{4}\phi^4}_{f(\phi)} + \frac{\alpha^2}{4\beta} = \frac{1}{4\beta} (\alpha + \beta\phi^2)^2, \quad (7)$$

so that the integral becomes:

$$\int_0^{\phi(x)} \frac{d\phi}{\alpha + \beta\phi^2} = \sqrt{\frac{1}{2\kappa\beta}} x \quad \Rightarrow \quad \frac{1}{\alpha} \int_0^{\phi(x)} \frac{d\phi}{1 - \frac{\beta}{-\alpha}\phi^2} = \sqrt{\frac{1}{2\kappa\beta}} x \quad (8)$$

Note that $\alpha < 0$. Using $\frac{d}{dx} \tanh^{-1}(x) = \frac{1}{1-x^2}$, we can then solve this integral to get:

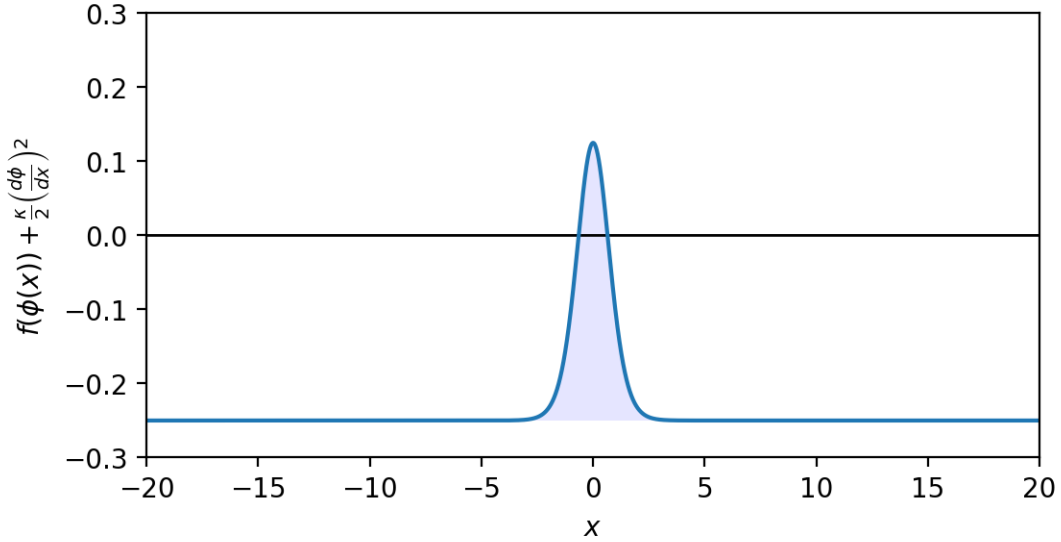
$$\phi(x) = \sqrt{\frac{-\alpha}{\beta}} \tanh\left(\frac{x}{\xi}\right), \quad \text{where} \quad \xi = \sqrt{\frac{2\kappa}{-\alpha}} \quad (9)$$

is the interfacial width.

1.2.2 Surface tension γ

In this subsection, we will introduce the concept of surface tension γ and derive its expression for a flat interface. First, let us consider the energy density $f(\phi) + \frac{\kappa}{2} \left(\frac{d\phi}{dx} \right)^2$, which consists of the local and semi-local terms.

```
[ ]: fig, ax = plt.subplots(figsize=(6,3))
ax.set_xlabel('$x$')
ax.
    ↪set_ylabel('$f(\phi(x))+\frac{\kappa}{2}\left(\frac{d\phi}{dx}\right)^2$')
ax.set_xlim(-20,20)
ax.set_ylim(-0.3,0.3)
ax.plot(x, x*0, c='black', linewidth=1)
ax.plot(x, f_density)
ax.fill_between(x, f_density, -alpha**2/(4*beta)*np.ones(np.shape(x)),
    ↪facecolor='blue', alpha=0.1)
plt.show()
```



The plot above shows the energy density $f(\phi(x)) + \frac{\kappa}{2} \left(\frac{d\phi}{dx} \right)^2$ as a function of x for the density profile $\phi(x)$, shown in the previous plot. In the bulk, the energy density is mostly flat and negative, which is equal to $f\left(\sqrt{\frac{-\alpha}{\beta}}\right) = -\frac{\alpha^2}{4\beta}$. Across the interface, the energy density goes up and then comes back to its bulk value. The interfacial energy is defined to be excess energy across this interface, *i.e.*, the shaded region in the plot above. More specifically, the interfacial energy is defined to be:

$$H_{\text{interface}} = A \int_{-\infty}^{\infty} \left\{ f(\phi(x)) + \frac{\kappa}{2} \left(\frac{d\phi}{dx} \right)^2 + \frac{\alpha^2}{4\beta} \right\} dx. \quad (10)$$

Now we can refer to the **magic equation** which we derived above, *i.e.*:

$$f(\phi) - \frac{\kappa}{2} \left(\frac{d\phi}{dx} \right)^2 = C = -\frac{\alpha^2}{4\beta} \quad (11)$$

Substituting this to $F_{\text{interface}}$, the interfacial energy becomes:

$$H_{\text{interface}} = A\kappa \int_{-\infty}^{\infty} \left(\frac{d\phi}{dx} \right)^2 dx. \quad (12)$$

Finally the surface tension γ is defined to be the interfacial energy per interfacial area. Thus we get:

$$\gamma = \kappa \int_{-\infty}^{\infty} \left(\frac{d\phi}{dx} \right)^2 dx. \quad (13)$$

Substituting $\phi(x) = \sqrt{\frac{-\alpha}{\beta}} \tanh\left(\frac{x}{\xi}\right)$, where $\xi = \sqrt{\frac{2\kappa}{-\alpha}}$, to the above equation, we get the formula for the surface tension for this specific energy:

$$\gamma = \sqrt{\frac{-8\kappa\alpha^3}{9\beta^2}}. \quad (14)$$

1.3 Curved interface ($d > 1$ dimension)

We will now go back to $d > 1$ dimensional problems. In general, the interface between the liquid and the gas phase is not always flat but often curved.

1.3.1 Macroscopic picture of surface tension

Let us consider a liquid droplet in three-dimension and far from the boundaries (such as solid walls). The total energy of the liquid droplet has two contributions:

$$H = H_{\text{bulk}} + H_{\text{interface}}. \quad (15)$$

The first contribution comes the bulk energy F_{bulk} , which is equal to the local energy density $f(\phi)$ multiplied by the volume of the droplet V :

$$H_{\text{bulk}} \simeq f(\phi_l)V = \text{constant}. \quad (16)$$

Note that we have assumed the density of the liquid inside the droplet, ϕ_l , to be constant throughout the bulk of the droplet. Consequently, the volume of the liquid droplet is also constant by conservation of mass. Thus F_{bulk} is constant. The second contribution to F comes from the interfacial energy $F_{\text{interface}}$ which is equal to the surface tension γ multiplied by the surface area of the droplet S :

$$H_{\text{interface}} \simeq \gamma S. \quad (17)$$

Since $\gamma > 0$, the equilibrium shape of the liquid droplet, which minimizes the total energy H , is therefore a sphere.

The surface tension γ can also be thought as a force per unit length. To illustrate this, let us consider a thin film of soap, as shown in the picture below. Technically, a soap film is an air-liquid-air interface, so the surface tension would be doubled. Let us imagine that this two-dimensional soap film is confined between three fixed rods (black colour in the picture below) and a movable rod which can move laterally along x (red colour in the picture below). The energy stored inside this soap film is then:

$$H_{\text{interface}} = \gamma S = \gamma \ell x. \quad (18)$$

Therefore the force by the soap film on the red rod is equal to:

$$F_{\text{interface}} = -\frac{dH_{\text{interface}}}{dx} = -\gamma \ell. \quad (19)$$

$F_{\text{interface}}$ is negative since the direction is to the left in the picture below. Thus a surface tension γ is a force (per unit length) which tends to minimize the surface area of the interface.

```
[ ]: fig, ax = plt.subplots(figsize=(6,4))

x = np.arange(1, 4, 0.001)
x1 = np.arange(1, 3, 0.001)

ax.set_xlim(0, 5)
ax.set_ylim(-2, 2)
ax.set_aspect(1)
ax.set_xticks([])
ax.set_yticks([])

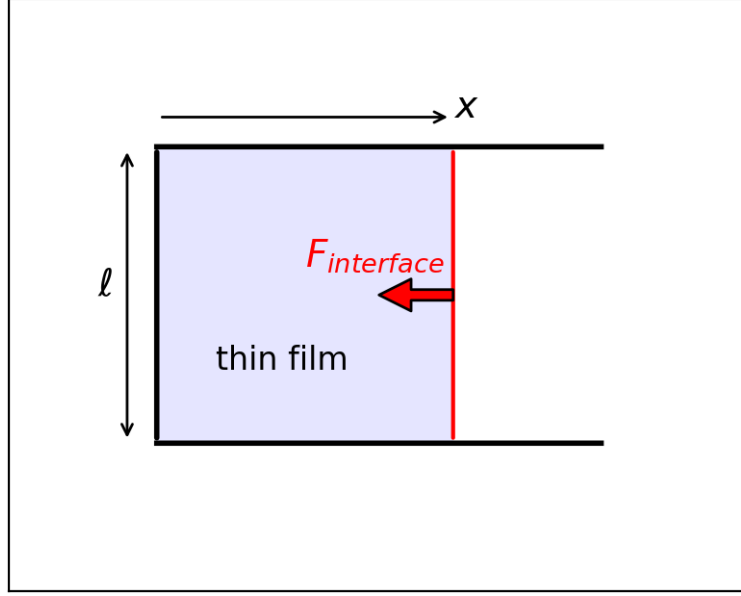
ax.fill_between(x1, np.ones(len(x1))*-1, np.ones(len(x1))*1, facecolor='blue',
    ↪alpha=0.1)

ax.plot(x, np.ones(len(x))*-1, c='black', linewidth=2)
ax.plot(x, np.ones(len(x))*1, c='black', linewidth=2)

ax.annotate('', xy=(1,-1), xytext=(1,1), arrowprops=dict(edgecolor='black',
    ↪arrowstyle='-', linewidth=2))
ax.annotate('', xy=(3,-1), xytext=(3,1), arrowprops=dict(edgecolor='red',
    ↪arrowstyle='-', linewidth=1.5))
ax.annotate('', xy=(0.8,-1), xytext=(0.8,1), arrowprops=dict(edgecolor='black',
    ↪arrowstyle='<->'))
ax.annotate('', xy=(3,1.2), xytext=(1,1.2), arrowprops=dict(edgecolor='black',
    ↪arrowstyle='->'))
ax.annotate('', xy=(2.5,0.0), xytext=(3,0.0),
    ↪arrowprops=dict(edgecolor='black', facecolor='red'))

ax.annotate('$\\ell$', c='black', fontsize=14, xy=(0.6,0.0))
ax.annotate('$x$', c='black', fontsize=14, xy=(3,1.2))
ax.annotate('$F_{interface}$', c='red', fontsize=14, xy=(2.0,0.2))
ax.annotate('thin film', c='black', fontsize=12, xy=(1.4,-0.5))

plt.show()
```

1.3.2 Stress tensor

Consider a smooth surface S , which can be open or closed. For an open surface, we can arbitrarily define one side of the surface to be the *inner* side and the other side to be the *outer* side. For a closed surface, on the other hand, the inner side is always defined to be the volume which is enclosed by the surface. At any point P on the surface, we may then define the outward normal unit vector $\hat{\mathbf{n}}$ to be perpendicular to S and point in the outer direction.

Let us consider some surface element $d\mathbf{S} = dS \hat{\mathbf{n}}$ inside some fluid, as depicted in the picture below. The direction of $d\mathbf{S}$ is defined to be the outward unit normal, which in this case is assumed to point to the right. The differential force acting on the inner fluid by the outer fluid (or by the boundary if it is a solid wall on the other side) is then given by:

$$dF_\alpha = \sigma_{\alpha\beta} dS_\beta, \quad (20)$$

where $\sigma_{\alpha\beta}(\mathbf{r})$ is the surface tensor, which is evaluated at a point on this surface element. In the case of an isotropic stress, the stress tensor can be written as:

$$\sigma_{\alpha\beta}(\mathbf{r}) = -p(\mathbf{r})\delta_{\alpha\beta}, \quad (21)$$

where $p(\mathbf{r})$ is called the pressure of the fluid at point \mathbf{r} in space. Notice the negative sign. If we have a fluid inside a cubic box of size $L \times L \times L$, the fluid will push each side of the box, with force magnitude pL^2 (if p is positive).

Let us consider a fluid inside some volume V . V can be enclosed by other fluid or solid walls. Let us denote $\mathbf{f}(\mathbf{r})$ to be the force density (force per unit volume) acting on the fluid at \mathbf{r} . The net force acting on the fluid (by the surrounding fluid or the solid boundaries) is then given by the integral of \mathbf{f} over the whole fluid volume:

$$\mathbf{F} = \int_V \mathbf{f} dV. \quad (22)$$

However, from the definition of the stress tensor, the net force acting on the fluid by the boundary can also be written as:

$$F_\alpha = \oint_{\partial V} \sigma_{\alpha\beta} dS_\beta. \quad (23)$$

Thus,

$$\int_V f_\alpha dV = \oint_{\partial V} \sigma_{\alpha\beta} dS_\beta = \int_V \partial_\beta \sigma_{\alpha\beta} dV, \quad (24)$$

and therefore,

$$f_\alpha = \partial_\beta \sigma_{\alpha\beta}. \quad (25)$$

```
[ ]: fig, ax = plt.subplots(figsize=(6,4))

x = np.arange(0, 5, 0.001)
x1 = np.arange(2, 2.5, 0.001)

ax.set_xlim(0, 5)
ax.set_ylim(-2, 2)
ax.set_aspect(1)
ax.set_xticks([])
ax.set_yticks([])

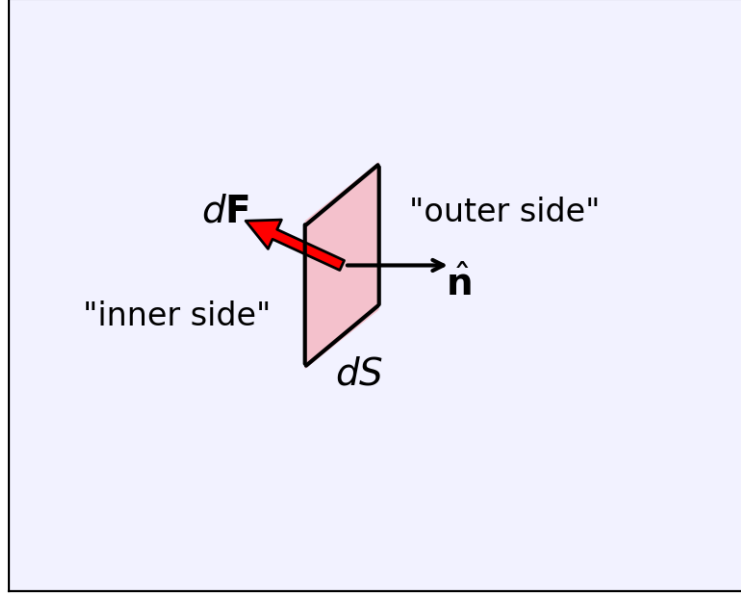
ax.annotate('', xy=(1.6,0.5), xytext=(2.25,0.2),
    ↪arrowprops=dict(edgecolor='black', facecolor='red'))

ax.fill_between(x, np.ones(len(x))*-2, np.ones(len(x))*2, facecolor='blue',
    ↪alpha=0.05)
ax.fill_between(x1, 0.8*x1 - 2.1, 0.8*x1 - 1.1, facecolor='red', alpha=0.2)

ax.annotate('', xy=(2,-0.5), xytext=(2,0.51),
    ↪arrowprops=dict(edgecolor='black', arrowstyle='-', linewidth=1.5))
ax.annotate('', xy=(2.5,-0.1), xytext=(2.5,0.9),
    ↪arrowprops=dict(edgecolor='black', arrowstyle='-', linewidth=1.5))
ax.annotate('', xy=(1.98,0.45), xytext=(2.52,0.9),
    ↪arrowprops=dict(edgecolor='black', arrowstyle='-', linewidth=1.5))
ax.annotate('', xy=(1.98,-0.5), xytext=(2.52,-0.05),
    ↪arrowprops=dict(edgecolor='black', arrowstyle='-', linewidth=1.5))
ax.annotate('', xy=(3,0.2), xytext=(2.25,0.2),
    ↪arrowprops=dict(edgecolor='black', arrowstyle='->', linewidth=1.5))

ax.annotate('$\hat{\mathbf{n}}$', c='black', fontsize=14, xy=(2.95,0.0))
ax.annotate('$dS$', c='black', fontsize=14, xy=(2.2,-0.6))
ax.annotate('$d\mathbf{F}$', c='black', fontsize=14, xy=(1.3,0.5))
ax.annotate('"inner side"', c='black', fontsize=12, xy=(0.5,-0.2))
ax.annotate('"outer side"', c='black', fontsize=12, xy=(2.7,0.5))

plt.show()
```



Now we will derive the formula for the *elastic* stress tensor for our Hamiltonian:

$$H[\phi] = \int_V \underbrace{\left\{ \frac{\alpha}{2} \phi^2 + \frac{\kappa}{2} \phi^4 + \frac{\kappa}{2} |\nabla \phi|^2 \right\}}_{g(\phi, \partial_\alpha \phi)} dV. \quad (26)$$

Suppose that we have some fluid, confined to a trapezoidal box of volume V in space, as we can see in the picture below. Now we can deform the fluid *affinely* (e.g. by shearing the box) through some infinitesimal strain $\delta \mathbf{r}$ such that $\mathbf{r} \rightarrow \mathbf{r} + \delta \mathbf{r}$. Under this *affine* deformation, the fluid density $\phi(\mathbf{r})$ and the volume of the box V transform as:

$$\phi(\mathbf{r}) \rightarrow \phi(\mathbf{r} - \delta \mathbf{r}) = \phi(\mathbf{r}) + \delta \phi(\mathbf{r}) \quad \text{and} \quad V \rightarrow V + \delta V. \quad (27)$$

In other words, affine deformation simply translates the value of ϕ through space by $\delta \mathbf{r}$. By Taylor expanding $\phi(\mathbf{r} - \delta \mathbf{r})$, we can find:

$$\delta \phi = -\delta \mathbf{r} \cdot \nabla \phi. \quad (28)$$

Now we can calculate the change in the total energy due to this affine deformation:

$$\delta H = H[\phi + \delta \phi] - H[\phi] \quad (29)$$

$$= \int_{V+\delta V} g(\phi + \delta \phi, \partial_\alpha \phi + \delta \partial_\alpha \phi) dV - \int_V g(\phi, \partial_\alpha \phi) dV \quad (30)$$

$$= \int_V g(\phi + \delta \phi, \partial_\alpha \phi + \delta \partial_\alpha \phi) dV + \int_{\delta V} g(\phi + \delta \phi, \partial_\alpha \phi + \delta \partial_\alpha \phi) dV - \int_V g(\phi, \partial_\alpha \phi) dV \quad (31)$$

Now we can Taylor expand:

$$g(\phi + \delta \phi, \partial_\alpha \phi + \delta \partial_\alpha \phi) = g(\phi, \partial_\alpha \phi) + \delta \phi \frac{\partial g}{\partial \phi} + (\delta \partial_\alpha \phi) \frac{\partial g}{\partial (\partial_\alpha \phi)} \quad (32)$$

In particular, the second integral becomes:

$$\int_{\delta V} g(\phi + \delta\phi, \partial_\alpha\phi + \delta\partial_\alpha\phi) dV \simeq \int_{\delta V} g(\phi, \partial_\alpha\phi) dV, \quad (33)$$

where we have neglected order $\sim \delta^2$. Thus the change in the total energy is

$$\delta H = \int_V \left\{ \delta\phi \frac{\partial g}{\partial \phi} + (\partial_\alpha \delta\phi) \frac{\partial g}{\partial (\partial_\alpha \phi)} \right\} dV + \int_{\delta V} g(\phi, \partial_\alpha \phi) dV \quad (34)$$

$$= \int_V \left\{ \delta\phi \frac{\partial g}{\partial \phi} + (\partial_\alpha \delta\phi) \frac{\partial g}{\partial (\partial_\alpha \phi)} \right\} dV + \oint_{\partial V} g(\phi, \partial_\alpha \phi) \delta \mathbf{r} \cdot d\mathbf{S}. \quad (35)$$

Note that when you displace a surface element $d\mathbf{S}$ by $\delta \mathbf{r}$, the volume covered by this travelling surface element is $\delta \mathbf{r} \cdot d\mathbf{S}$. Next we can use the integration by parts on the first term:

$$\delta H = \int_V \left\{ \frac{\partial g}{\partial \phi} - \partial_\alpha \left(\frac{\partial g}{\partial (\partial_\alpha \phi)} \right) \right\} \delta\phi dV + \oint_{\partial V} \frac{\partial g}{\partial (\partial_\alpha \phi)} \delta\phi dS_\alpha + \oint_{\partial V} g \delta r_\alpha dS_\alpha \quad (36)$$

$$= - \int_V \mu \delta r_\alpha \partial_\alpha \phi dV - \oint_{\partial V} \frac{\partial g}{\partial (\partial_\alpha \phi)} \delta r_\beta \partial_\beta \phi dS_\alpha + \oint_{\partial V} g \delta r_\alpha dS_\alpha \quad (37)$$

$$= \int_V \phi \partial_\alpha (\mu \delta r_\alpha) dV - \oint_{\partial V} \phi \mu \delta r_\alpha dS_\alpha + \oint_{\partial V} \left\{ g \delta_{\alpha\beta} - (\partial_\alpha \phi) \frac{\partial g}{\partial (\partial_\beta \phi)} \right\} \delta r_\alpha dS_\beta \quad (38)$$

$$= \oint_{\partial V} \left\{ (g - \phi \mu) \delta_{\alpha\beta} - (\partial_\alpha \phi) \frac{\partial g}{\partial (\partial_\beta \phi)} \right\} \delta r_\alpha dS_\beta + \int_V (\phi \partial_\alpha \mu) \delta r_\alpha dV \quad (39)$$

In the last line we assumed the fluid to be incompressible, *i.e.* $\partial_\alpha \delta r_\alpha = 0$. Using the first law of thermodynamics, the change in the energy is also equal to:

$$\delta H = \delta W - \delta Q, \quad (40)$$

where δQ is the heat dissipated into the environment, which is zero since affine deformation is frictionless. δW is the work done on the system by the external forces. In order to apply this affine deformation to the system, we must exert: 1. some external surface force $\mathbf{F}^{\text{surface}}$ to the walls (using our hands), and/or 2. some external body force density \mathbf{f}^{body} to the bulk of the fluid (such as gravity and electric field).

Let us consider the external surface force first. The force acting on the walls by the external force (such as our hands) is $\mathbf{F}^{\text{surface}}$. The force acting on the walls by the fluid is $-\oint_{\partial V} \sigma_{\alpha\beta} dS_\beta$. Since we assume mechanical equilibrium throughout the deformation process, the net force on the walls has to be zero:

$$F_\alpha^{\text{surface}} - \oint_{\partial V} \sigma_{\alpha\beta} dS_\beta = 0, \quad \text{for } \alpha = x, y, z. \quad (41)$$

Similar principle also applies for the external body force. Let's consider some fluid element dV inside the bulk of the fluid. The force acting on this fluid element by the external force (such as gravity and electric field) is $\mathbf{f}^{\text{body}} dV$. The force acting on this fluid element by the surrounding fluid is $\partial_\beta \sigma_{\alpha\beta} dV$. Since this fluid element is always in mechanical equilibrium, we also must have the force balance:

$$f_\alpha^{\text{body}} + \partial_\beta \sigma_{\alpha\beta} = 0, \quad \text{for } \alpha = x, y, z. \quad (42)$$

Thus the work done by the external forces is $\delta W = \mathbf{F} \cdot \delta \mathbf{r}$, or:

$$\delta W = \oint_{\partial V} \sigma_{\alpha\beta} \delta r_\alpha dS_\beta - \int_V (\partial_\beta \sigma_{\alpha\beta}) \delta r_\alpha dV = \int_V \underbrace{\sigma_{\alpha\beta}}_{\text{stress}} \underbrace{\partial_\beta \delta r_\alpha}_{\text{strain}} dV \quad (43)$$

Note that $\partial_\beta \delta r_\alpha$ is also called the strain tensor. Now we can then equate $\delta H = \delta W$. Comparing the surface term, we get the elastic stress tensor:

$$\sigma_{\alpha\beta} = \underbrace{(g - \phi\mu)}_{-p} \delta_{\alpha\beta} - (\partial_\alpha \phi) \frac{\partial g}{\partial (\partial_\beta \phi)}, \quad (44)$$

where we have also identified the isotropic pressure to be $p = \phi\mu - g$. Equating the volume term, we get the elastic force density:

$$\mathbf{f} = -\phi \nabla \mu. \quad (45)$$

One can also verify that $f_\alpha = \partial_\beta \sigma_{\alpha\beta}$.

```
[ ]: fig, ax = plt.subplots(figsize=(8,4))

x = np.arange(0.5, 2.5, 0.001)
x1 = np.arange(3.5, 4, 0.001)
x2 = np.arange(4, 5.5, 0.001)
x3 = np.arange(5.5, 6, 0.001)
x4 = np.arange(3.5, 5.5, 0.001)
x5 = np.arange(4, 6, 0.001)

ax.set_xlim(0, 6.5)
ax.set_ylim(-1.5, 1.5)
ax.set_aspect(1)
ax.set_xticks([])
ax.set_yticks([])

ax.fill_between(x, np.ones(len(x))*-1, np.ones(len(x))*1, facecolor='blue',
    ↪alpha=0.1)
ax.fill_between(x1, np.ones(len(x1))*-1, 4*x1-15, facecolor='blue', alpha=0.1)
ax.fill_between(x2, np.ones(len(x2))*-1, np.ones(len(x2))*1, facecolor='blue',
    ↪alpha=0.1)
ax.fill_between(x3, 4*x3-23, np.ones(len(x3))*1, facecolor='blue', alpha=0.1)

ax.plot(x, np.ones(len(x))*-1, c='black', linewidth=2)
ax.plot(x, np.ones(len(x))*1, c='black', linewidth=2)
ax.plot(x4, np.ones(len(x4))*-1, c='black', linewidth=2)
ax.plot(x5, np.ones(len(x5))*1, c='black', linewidth=2)

ax.annotate('', xy=(0.5,-1), xytext=(0.5,1), arrowprops=dict(edgecolor='black',
    ↪arrowstyle='-', linewidth=2))
ax.annotate('', xy=(2.5,-1), xytext=(2.5,1), arrowprops=dict(edgecolor='black',
    ↪arrowstyle='-', linewidth=2))
ax.annotate('', xy=(4,1), xytext=(3.5,-1), arrowprops=dict(edgecolor='black',
    ↪arrowstyle='-', linewidth=2))
ax.annotate('', xy=(6,1), xytext=(5.5,-1), arrowprops=dict(edgecolor='black',
    ↪arrowstyle='-', linewidth=2))
```

```

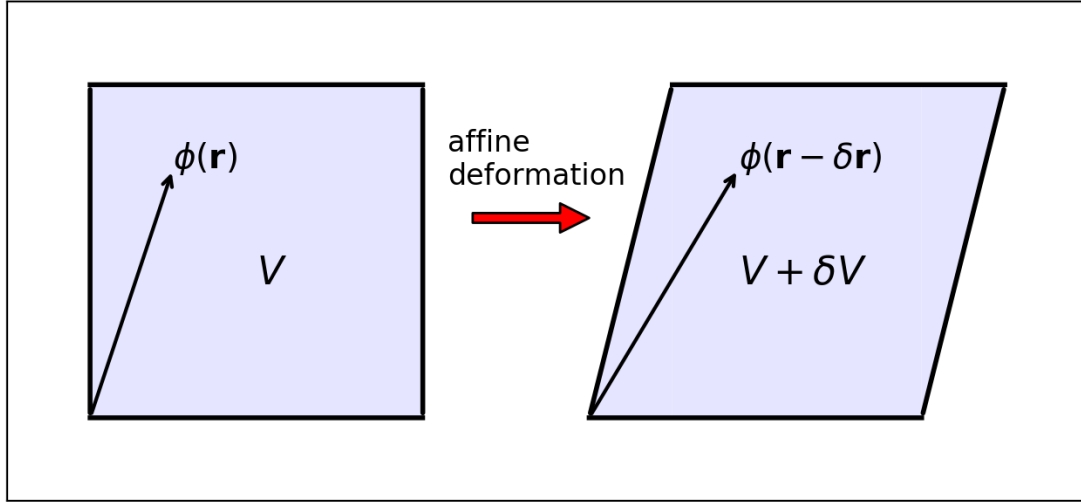
ax.annotate('', xy=(1,0.5), xytext=(0.5,-1), arrowprops=dict(edgecolor='black',
↪arrowstyle='->', linewidth=1.5))
ax.annotate('', xy=(4.4,0.5), xytext=(3.5,-1),
↪arrowprops=dict(edgecolor='black', arrowstyle='->', linewidth=1.5))

ax.annotate('', xy=(3.5,0.2), xytext=(2.8,0.2),
↪arrowprops=dict(edgecolor='black', facecolor='red'))

ax.annotate('$\phi(\mathbf{r})$', c='black', fontsize=14, xy=(1,0.5))
ax.annotate('$\phi(\mathbf{r})-\delta\mathbf{r})$', c='black', fontsize=14,
↪xy=(4.4,0.5))
ax.annotate('$V$', c='black', fontsize=16, xy=(1.5,-0.2))
ax.annotate('$V+\delta V$', c='black', fontsize=16, xy=(4.4,-0.2))
ax.annotate('affine \ndeformation', c='black', fontsize=12, xy=(2.65,0.4))

plt.show()

```



1.3.3 Convection vs diffusion

Thus we have derived the formula for the elastic stress for the given Hamiltonian. Under affine deformation, the fluid exerts some elastic stress to the walls, and also to each other inside the bulk. However for a general deformation, this is not the only stress present in the fluid. We also have a viscous stress, whose coefficient is called the viscosity, which is responsible for heat dissipation.

One might also notice that the presence or absence of a mechanical stress does not affect the Cahn-Hilliard equation:

$$\frac{\partial \phi}{\partial t} = M \nabla^2 \frac{\delta H}{\delta \phi} \quad (\text{Cahn-Hilliard equation}) \quad (46)$$

This is because Cahn-Hilliard equation only describes the diffusion process in the binary fluid. In

general, there are two ways a fluid can move inside the system: 1) diffusion and 2) convection.

An example of diffusion process is when we put a drop of red ink inside a static and very viscous solvent. This ink will spread slowly until the whole fluid becomes homogenously red. This diffusion process is due to the Brownian motion of the ink molecules. Typically the higher the temperature is, the faster the diffusion process will be.

An example of convection is when we put a beach ball on a flowing river. This beach ball will move with the same velocity as the water. If we put a drop of red ink on a flowing river, this blob of red ink will also move with the same velocity as the water, while at the same time diffuses around its centre of mass. To describe the convection process in a binary fluid, we need to define another field, which is the velocity field $\mathbf{u}(\mathbf{r}, t)$ (in addition to $\phi(\mathbf{r}, t)$). The time evolution of $\mathbf{u}(\mathbf{r}, t)$ will be described by the Navier-Stokes equation:

$$\underbrace{\rho}_{\text{mass density}} \underbrace{\left(\frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla \right)}_{\text{acceleration}} u_\alpha = \underbrace{\partial_\beta \sigma_{\alpha\beta}}_{\text{force density}} . \quad (\text{Navier-Stokes equation}) \quad (47)$$

In essence, the Navier-Stokes equation is just a momentum conservation or Newton's second law. Thus to take into account of both diffusion and convection process in a binary fluid properly, we need to solve the coupled Cahn-Hilliard-Navier-Stokes equation:

$$\frac{\partial \phi}{\partial t} + (\mathbf{u} \cdot \nabla) \phi = M \nabla^2 \frac{\delta H}{\delta \phi} \quad (\text{Cahn-Hilliard equation}) \quad (48)$$

$$\rho \left(\frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla \right) u_\alpha = \partial_\beta \sigma_{\alpha\beta} \quad (\text{Navier-Stokes equation}) \quad (49)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (\text{incompressibility condition}). \quad (50)$$

Notice that the first two equations are coupled through the stress tensor $\sigma_{\alpha\beta}$, which depends on ϕ , and the convection term $(\mathbf{u} \cdot \nabla) \phi$.

In practice, we can also define the diffusion timescale $t_{\text{diffusion}}$ and the convection timescale $t_{\text{convection}}$ to be:

$$t_{\text{diffusion}} = \frac{L^2}{M\alpha} \quad \text{and} \quad t_{\text{convection}} = \frac{L}{U}, \quad (51)$$

where L is the system size and U is the typical magnitude of the fluid velocity in the system (such the velocity of the flowing river under gravity). If $t_{\text{diffusion}} \ll t_{\text{convection}}$, then the diffusion process dominates over convection and the problem can be simply described by Cahn-Hilliard equation alone.

1.3.4 Laplace pressure

Let us consider a high density phase (*e.g.* liquid) in coexistence with a low density phase (*e.g.* gas) inside our system. We must also have an interface, which separates the liquid from the gas phase. There is an energy cost to create this interface (which is the surface tension γ), and thus the equilibrium state always favours the one with minimal interfacial area, while preserving the total volume of the liquid and the gas phase in the system. If we have roughly equal volume of liquid and gas phase, the interface will be flat, if not, the interface will be spherical. Let us also denote ϕ_+ to be the density of the liquid in the bulk and ϕ_- to be the density of the gas in the bulk. For

a flat interface, ϕ_{\pm} will correspond to the minima of the bulk energy density $f(\phi) = \frac{\alpha}{2}\phi^2 + \frac{\beta}{4}\phi^4$ (or binodals):

$$\phi_{\pm} = \pm\phi_B, \quad \text{where } \phi_B = \sqrt{\frac{-\alpha}{\beta}} \text{ (for flat interface).} \quad (52)$$

However, for a curved interface, such as a liquid droplet surrounded by the gas phase, ϕ_{\pm} will be slightly shifted away from the binodal values, *i.e.*:

$$\phi_{\pm} = \pm\phi_B + \epsilon_{\pm} \text{ (for curved interface).} \quad (53)$$

The reason for this is because of the Laplace pressure: the pressure inside the droplet will be slightly higher than that outside. To illustrate this, let us consider a spherical liquid droplet of radius R surrounded by gas phase, as shown in the figure below. Let us denote P_+ and P_- to be the pressure inside and outside the droplet respectively. Now let us consider the upper hemispherical surface. There are two forces acting on this hemispherical surface. The first one is due to the pressure difference $\Delta P = P_+ - P_-$. The force acting on the upper hemispherical surface is equal to $\Delta P \pi R^2$ in the upwards direction. The second force is due to the surface tension γ : the upper hemispherical surface is pulled by the lower hemisphere right at the equator. The magnitude of this force is $\gamma 2\pi R$ in the direction downwards. Thus balancing the two forces, we get:

$$P_+ - P_- = \frac{2\gamma}{R}. \quad (54)$$

Note that for d -dimensional spherical droplet, the Laplace pressure is given by:

$$P_+ - P_- = \frac{\gamma(d-1)}{R}. \quad (55)$$

Apart from the force balance, the chemical potential inside the droplet also has to be the same as that outside the droplet. If not, there will be a diffusive current from high chemical potential regions to low chemical potential regions. Equating the two chemical potentials, we get:

$$\mu(\phi_+) = \mu(\phi_-). \quad (56)$$

Now from the previous section, the chemical potential is given by the formula:

$$\mu = f'(\phi) - \kappa \nabla^2 \phi. \quad (57)$$

However since we are only interested in the value of μ in the bulk (where ϕ is uniform), we can ignore the gradient in ϕ , and thus we get:

$$f'(\phi_+) = f'(\phi_-) \quad \Rightarrow \quad f'(\phi_B + \epsilon_+) = f'(-\phi_B + \epsilon_-). \quad (58)$$

Next, we can Taylor expand:

$$f'(\phi_B + \epsilon_+) = f''(\phi_B)\epsilon_+ + \mathcal{O}(\epsilon^2) \quad (59)$$

$$f'(-\phi_B + \epsilon_-) = f''(\phi_B)\epsilon_- + \mathcal{O}(\epsilon^2). \quad (60)$$

Note that since $f(\phi)$ is symmetric, $f''(\phi_B) = f''(-\phi_B)$. Thus the equality of chemical potential implies that $\epsilon_+ = \epsilon_-$, *i.e.* the values of ϕ_{\pm} are shifted by the same amount from the binodals.

Now let us go back to the Laplace pressure. From the previous section, we have the formula for the pressure:

$$P = \phi\mu - g, \text{ where } g = f(\phi) + \frac{\kappa}{2}|\nabla\phi|^2 \text{ and } \mu = f'(\phi) - \kappa\nabla^2\phi. \quad (61)$$

Since we are only interested in the value of P in the bulk, we can ignore the gradient terms to get:

$$P(\phi) = \phi f'(\phi) - f(\phi). \quad (62)$$

Now substituting this to the Laplace pressure equation, we get:

$$\phi_+ f'(\phi_+) - \phi_- f'(\phi_-) - f(\phi_+) + f(\phi_-) = \frac{2\gamma}{R} \quad (63)$$

$$(\phi_B + \epsilon) f'(\phi_B + \epsilon) - (-\phi_B + \epsilon) f'(-\phi_B + \epsilon) - f(\phi_B + \epsilon) + f(-\phi_B + \epsilon) = \frac{2\gamma}{R} \quad (64)$$

We can then Taylor expand for small ϵ and simplify to get

$$\epsilon = \frac{\gamma}{R\phi_B f''(\phi_B)}. \quad (65)$$

```
[ ]: fig, ax = plt.subplots(figsize=(4,4))

x = np.arange(-1, 1, 0.0001)

ax.set_xlim(-1.5, 1.5)
ax.set_ylim(-1.2, 1.8)
ax.set_aspect(1)
ax.set_xticks([])
ax.set_yticks([])

ax.fill_between(x, -np.sqrt(1-x*x), np.sqrt(1-x*x), facecolor='blue', alpha=0.1)
ax.plot(x, -np.sqrt(1-x*x), c='black', linewidth=2)
ax.plot(x, np.sqrt(1-x*x), c='black', linewidth=2)
ax.plot(x, -np.sqrt(1-x*x)/3, c='black', linewidth=1)

ax.annotate('', xy=(-1,-0.4), xytext=(-1,0), arrowprops=dict(edgecolor='red',
    ↪arrowstyle='->', linewidth=1.5))
ax.annotate('', xy=(1,-0.4), xytext=(1,0), arrowprops=dict(edgecolor='red',
    ↪arrowstyle='->', linewidth=1.5))
ax.annotate('', xy=(0,-0.4-0.32), xytext=(0,0-0.32),
    ↪arrowprops=dict(edgecolor='red', arrowstyle='->', linewidth=1.5))
ax.annotate('', xy=(0.4,-0.4-0.29), xytext=(0.4,0-0.29),
    ↪arrowprops=dict(edgecolor='red', arrowstyle='->', linewidth=1.5))
ax.annotate('', xy=(-0.4,-0.4-0.29), xytext=(-0.4,0-0.29),
    ↪arrowprops=dict(edgecolor='red', arrowstyle='->', linewidth=1.5))
ax.annotate('', xy=(0.75,-0.4-0.21), xytext=(0.75,0-0.21),
    ↪arrowprops=dict(edgecolor='red', arrowstyle='->', linewidth=1.5))
```

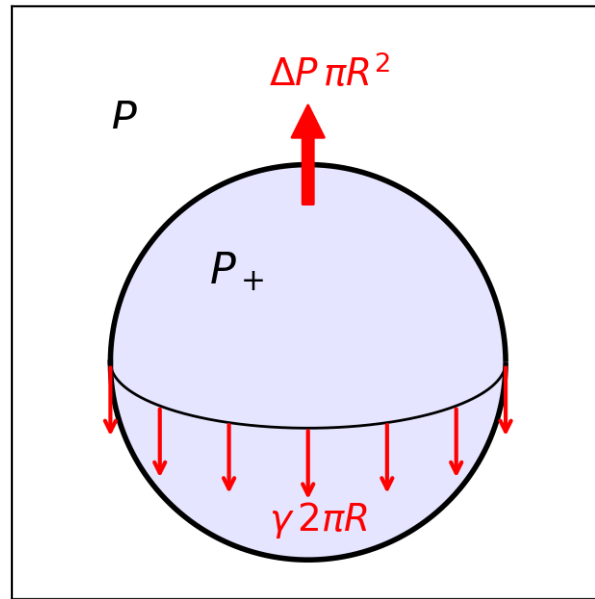
```

ax.annotate('', xy=(-0.75,-0.4-0.21), xytext=(-0.75,0-0.21),
    ↳arrowprops=dict(edgecolor='red', arrowstyle='->', linewidth=1.5))
ax.annotate('', xy=(0,1.3), xytext=(0,0.8), arrowprops=dict(edgecolor='red',
    ↳facecolor='red'))

ax.annotate('$P_+$', c='black', fontsize=16, xy=(-0.5,0.4))
ax.annotate('$P_{-}$', c='black', fontsize=16, xy=(-1,1.2))
ax.annotate('$\gamma \pi R$', c='red', fontsize=14, xy=(-0.2,-0.85))
ax.annotate('$\Delta P \pi R^2$', c='red', fontsize=14, xy=(-0.2,1.4))

plt.show()

```



1.3.5 Droplet evaporation

[]:

References: 1. L.D. Landau and E. M. Lifshitz, *Theory of Elasticity*, 3rd Edition (Elsevier, 1986). 2. M. E. Cates and E. Tjhung, Theory of binary fluid mixtures: from phase-separation kinetics to active emulsions, *J. Fluid Mech.* (2017). 3. T. Kruger, H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva, and E. M. Viggen, *The Lattice Boltzmann Method* (Springer, 2017)